



# ANNEX

(2876-X: OnTheBus)

## **Índex de continguts**

Annex A .....	4
Annex B: Proves i Resultats Simulats .....	9
Annex C: Programes generació proves Geolocalització .....	11



## Annex A

Taula sobre l'anàlisi de la disposició dels botons en els diferents terminals mòbils d'*Android*.

<u>Model del terminal</u>	<u>Botons físics</u>	<u>Botons tàctils</u>
HTC Dream	Trucada Home Menú Back Penjar trucada/encendre i apagar el terminal Volum Trackball Càmera Teclat Qwerty	
HTC Màgic	Trucada Home Menú Back Penjar trucada/encendre i apagar el terminal Volum Trackball	
HTC Hero	Trucada Home Menú Back Buscar Penjar trucada/encendre i apagar el terminal Volum Trackball	
HTC Nexus One	Volum Trackball	Back Menu

	Encendre i apagar el terminal	Home Buscar
Samsung Nexus S	Volum Encendre i apagar el terminal	Back Menu Buscar Home
HTC Wildfire	Volum Encendre i apagar el terminal	Home Menú Back Buscar Trackball Òptic
HTC Desire	Volum Encendre i apagar el terminal	Home Menú Back Buscar Trackball Òptic
HTC Wildfire S	Volum Encendre i apagar el terminal	Home Menú Back Buscar
HTC Desire S	Volum Encendre i apagar el terminal	Home Menú Back Buscar
HTC Desire Z	Volum Encendre i apagar el terminal Càmera Teclat Qwerty	Home Menú Back Buscar Trackball Tàctil
Samsung Galaxy S	Volum	Menú

	Encendre i apagar el terminal	Home Back
Samsung Galaxy S II	Volum Encendre i apagar el terminal	Menú Home Back
Samsung Galaxy mini	Volum Encendre i apagar el terminal	Menú Home Back
Samsung Galaxy Ace	Volum Encendre i apagar el terminal	Menú Home Back
Samsung Galaxy Pro	Volum Encendre i apagar el terminal Teclat Qwerty	Menú Home Back Buscar
Huawei U8110 (Ivy)	Volum Encendre i apagar el terminal Tecla de navegació "Trackball" Tecla OK Trucada Penjar Trucada Càmera	Home Menú Back
Huawei U8500	Trucada Menú Back Penjar Trucada Volum Encendre i apagar el terminal	Trackball òptic
Huawei U8650 (Sonic)	Volum	Home

	Encendre i apagar el terminal	Menú Back Buscar
<b>LG Optimus Me P350</b>	Tecla Enviar Encendre i apagar el terminal / Bloquejar Volum	Menú Home Back Buscar
<b>LG Optimus One P500</b>	Encendre i apagar el terminal / Bloquejar Volum Menú Home Back Buscar	
<b>LG Optimus 2X P990</b>	Encendre i apagar el terminal / Bloquejar Volum	Menú Home Back Buscar
<b>ZTE Blade</b>	Encendre i apagar el terminal / Bloquejar Volum Home Menú Back	
<b>Sony Ericsson Xperia X8</b>	Encendre i apagar el terminal / Bloquejar Volum Menú Home Back Càmera	
<b>Sony Ericsson Xperia X10</b>	Encendre i apagar el terminal / Bloquejar Volum Menú	

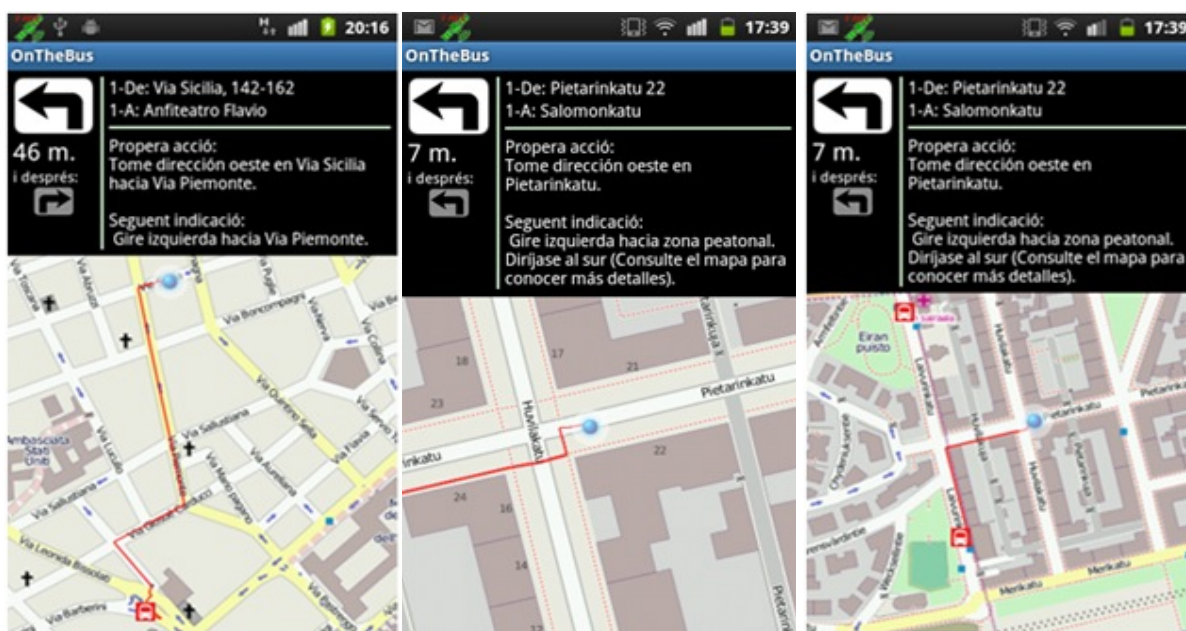
	Home Back Càmera	
<b>Sony Ericsson Xperia X10 mini Pro</b>	Encendre i apagar el terminal / Bloquejar Volum Menú Home Back Càmera Teclat Qwerty	
<b>Sony Ericsson Xperia Arc</b>	Encendre i apagar el terminal / Bloquejar Volum /Zoom Menú Home Back Càmera	
<b>Sony Ericsson Xperia Ray</b>	Encendre i apagar el terminal / Bloquejar Volum	Back Home Menu



## Annex B: Proves i Resultats Simulats

En aquest annex s'inclouen les diferents proves que s'han fet a les ciutats a les que no s'han pogut fer proves de camp. Per poder testejar els diferents mòduls s'utilitza *Fake GPS*, una aplicació disponible al *Android Market* i que permet establir una falsa localització GPS en les diferents aplicacions disponibles al mòbil, per tant, mitjançant aquest programa es pot simular que estem a qualsevol ciutat del món.

- **Rutes Walk**

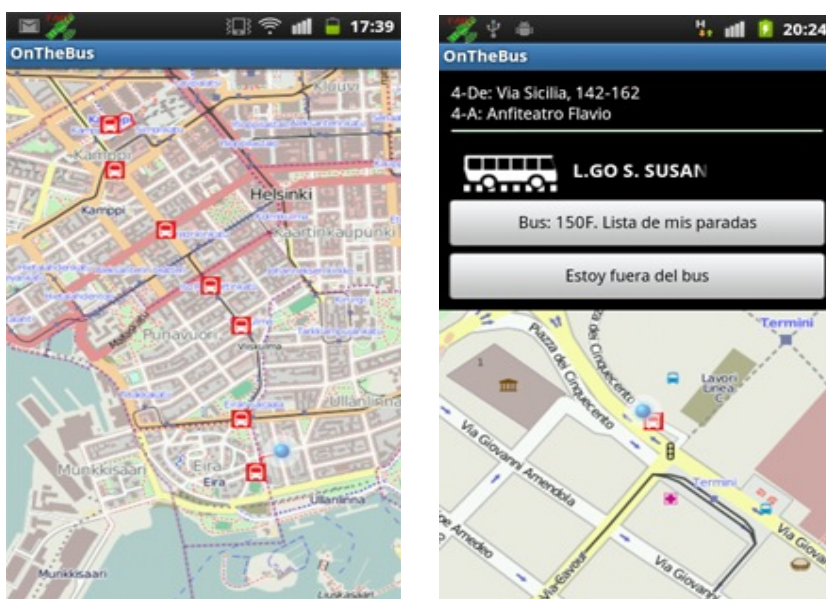


Il·lustració 1: Exemple a Roma;

Il·lustració 2: Ruta a Helsinki;

Il·lustració 3: Trajecte a Helsinki;

- **Rutes Bus**



Il·lustració 4: Bus a Helsinki;

Il·lustració 5: Exemple a Madrid;

- Rutes amb transbord

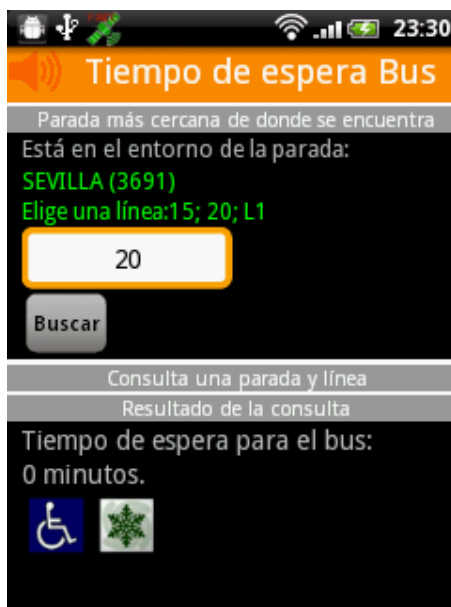


Il·lustració 6: Transbord a Madrid;

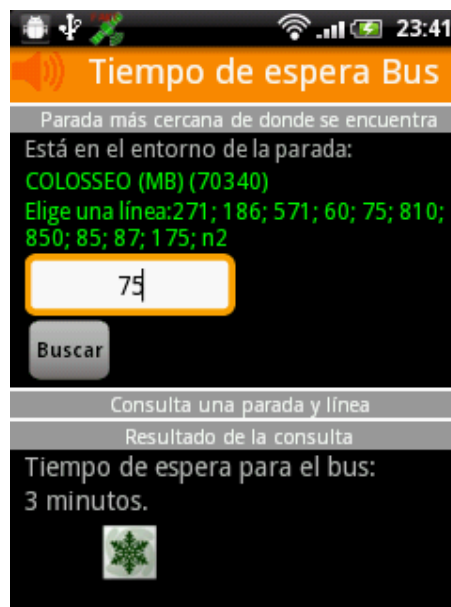


Il·lustració 7: Transbord a Roma;

- Temps d'Espera



Il·lustració 8: Temps Espera a Madrid;



Il·lustració 9: Temps Espera a Roma;

## Annex C: Programes generació proves Geolocalització

Programa utilitzat per a la obtenció de punts per simular un usuari caminant.

```
import java.text.DecimalFormat;

public class CreaRuta {
    public static void main(String[] args) {
        String [] unarray = {"41.389446,2.162225 41.384961,2.177273",
                             "41.384961,2.177273 41.384961,2.184273",
                             "41.384961,2.184273 41.354781,2.187211",
                             "41.354781,2.187211 41.368781,2.167211"
                             };

        for (int i = 0; i < unarray.length; i++) {
            DecimalFormat formato = new DecimalFormat("####.#####"); // Formato de salida
de dos decimales
            Double x1, x2, y1, y2;
            double hipo, nuevaX, nuevaY, tan, nuevaDistancia ;
            double DISTANCIA_INC = 0;
            int j = 1;
            String puntos [] = unarray[i].split(" ");
            String[] coordinates1 = puntos[0].split(",");
            String[] coordinates2 = puntos[1].split(",");

            x1 = new Double(coordinates1[1]);
            y1 = new Double(coordinates1[0]); //Coordenadas del 1er punto
            x2 = new Double(coordinates2[1]);
            y2 = new Double(coordinates2[0]); //Coordenadas del 2do punto
            hipo = Math.sqrt(Math.pow(x2-x1,2)+Math.pow(y2-y1,2)); //Calculo la distancia entre los
dos puntos = hipotenusa
            DISTANCIA_INC = hipo/40;
            tan = Math.atan((y2-y1)/(x2-x1)); // Radianes del Angulo Opuesto
            //System.out.println("Punto Inicial-> x: " + x1 + " , y: " + y1);
            //System.out.println("Punto Final-> x: " + x2 + " , y: " + y2 );
            //System.out.println("Distancia entro los Puntos: " + hipo );
            //System.out.println("Coordenas de (X,Y) de los puntos intermedios");
            //System.out.println("por la recta de los dos puntos desplazandose cada " +
DISTANCIA_INC + " unidades");
            //System.out.println("");
            // La nuevaDistancia se ira incrementando en 2 unidades, simulando la construcción de un
triangulo con una
            // hipotenusa de 2, 4, 6 hasta < a la distancia entre los dos puntos.
            for(nuevaDistancia = 0; nuevaDistancia < hipo; nuevaDistancia+=DISTANCIA_INC)
            {
                nuevaY = Math.sin(tan) * nuevaDistancia + y1; // Calculo de la Y a la nuevaDistancia
                nuevaX = Math.cos(tan) * nuevaDistancia + x1; // Calculo de la X a la nuevaDistancia
                // Asi obtenemos la nueva cordenada que pasa por la recta entre los dos punto
                System.out.println(formato.format(nuevaX) + "#" + formato.format(nuevaY));
            }
            //El ultimo punto lo mostramos directo
            //System.out.println("(f) Distancia: " + hipo + " | X: " + x2 + " | Y: " + y2);
        }
    }
}
```

Script perl que genera el fitxer KML que es pot carregar en el SDK d'Android.

```
#!/usr/bin/perl
```

```
open(OO,">salida.kml") or die "no se puede abrir fichero salida";
```

```
open(PP,"<putos.txt") or die "no se puede abrir el fichero";
```

```
print OO "<?xml version=\"1.0\" encoding=\"UTF-8\"?>\n";
```

```
print OO "<kml xmlns=\"http://earth.google.com/kml/2.x\">\n";
```

```
while (<PP>) {
```

```
    print $_;
```

```
    ($x,$y) = split("#",$_);
```

```
    $y =~ s/\n//g;
```

```
    $x =~ s/,/. /g;
```

```
    $y =~ s/,/. /g;
```

```
    print OO "\t<Placemark>\n";
```

```
    print OO "\t\t<name>$.</name>\n";
```

```
    print OO "\t\t<description></description>\n";
```

```
    print OO "\t\t<Point>\n";
```

```
    print OO "\t\t\t<coordinates>$x,$y,0</coordinates>\n";
```

```
    print OO "\t\t</Point>\n";
```

```
    print OO "\t</Placemark>\n";
```

```
}
```

```
print OO "</kml>";
```

```
close PP;
```

```
close OO;
```